The GridKa Tape System: status and outlook

Doris Ressmann[1,*], *Dorin* Lobontu[1], *Martin* Beitzinger[1], *Karin* Schaefer[1], *Andreas* Heiss[1], *Andreas* Petzold[1], and *Haykuhi* Musheghyan[1]

[1]Karlsruhe Institute of Technology

Abstract. Tape storage is still a cost effective way to keep large amounts of data over a long period of time and it is expected that this will continue in the future. The GridKa tape environment is a complex system of many hardware components and software layers. Configuring this system for optimal performance for all use cases is a non-trivial task and requires a lot of experience. We present the current status of the GridKa tape environment, report on recent upgrades and improvements and plans to further develop and enhance the system, especially with regard to the future requirements of the HEP experiments and their large data centers. The short-term planning mainly includes the transition from TSM to HPSS as the backend and the effects on the connection of dCache and xrootd. Recent changes of the vendor situation of certain tape technologies require a precise analysis of the impact and eventual adaptation of the mid-term planning, in particular with respect to scalability challenge that comes with HL-LHC on the horizon.

1 Introduction

Karlsruhe Institute of Technology (KIT) is divided into several locations around Karlsruhe. The main university is located in the city centre of Karlsruhe and is called south campus. There we have one IBM TS3500 library in use for our Backups. Around 10 km north of the city centre we have a second campus with many research institutions. This is called north campus. At this campus the Grid Computing Centre Karlsruhe (GridKa) is located. GridKa is the German Tier1 Centre of the Large Hardron Collider (LHC) , which is serving all four LHC experiments. At north campus we have currently in one building one IBM TS3500 library, and in our main building we have three SL8500 libraries from Oracle.

1.1 Tape Users

Access to KIT's tape storage is provided to three main user groups. The largest amount of tape storage is used by GridKa. Then there are also KIT users, who do their backup and archiving on our systems. Furthermore we have users of universities and research institutions of the state Baden Württemberg, who do their archiving within a project called BWDataArchive [1].

[*] e-mail: Doris.Ressmann@kit.edu

1.2 Tape Transfer Throughput

Providing high throughput data transfer to tape is a challenging task for majority of Tier-1 centers and this is no exception for GridKa. Solving challenging tasks is not easy and requires in-depth investigation of many factors. Some years ago there were many problems with LTO technologies mainly failures and instabilities. This is the reason why we changed to enterprise tape drive technology. However this experience is based on LTO4 and LTO5, and this might not be applicable any longer. Nevertheless we currently have no intention to migrate back to LTO technology. Additional information is provided in section 5. Another factor which significantly improved the bandwidth of GridKa data transfer to tape was the reconfiguration of our disk pools in front of the tape storage. Detailed information is provided in section 2. In addition, new monitoring tools were installed and configured (see section 4). As a result of all the changes, the GridKa tape has much better performance and higher reliability than before. The main goal is to retain and improve the existing setups.

2 Recent Improvements

Our Storage Area Network (SAN) setup was very complicated and error prone. Every dCache pool had a direct connection to each tape drive. In our current setup we have one proxy node for each virtual organisation (VO). Figure 1 displays the old and current network topology for the tape storage. This proxy node has a SAN connection to every single tape drive. However the storage pools only have a Local Area Network (LAN) connection to the proxy nodes. This new setup is very stable and reliable and as a result, a significant improvement in stability was achieved.

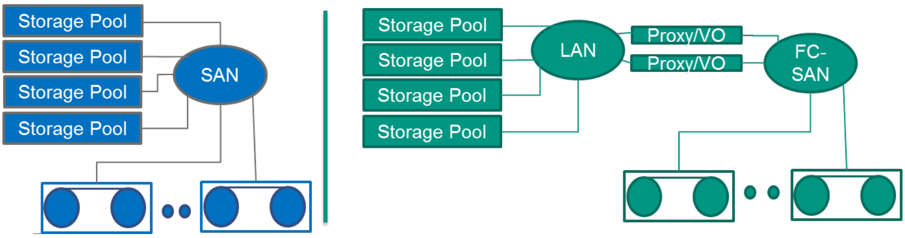

Figure 1: Our previous and current SAN setup

In figure 2 the improved tape recall rate by CMS is shown. This figure is a histogram comparing three month of tape usage with the old setup to the same amount of time with the current setup. With the old setup, the peak throughput for tape recall reached an average of around 50 MB/s, while with our current setup the peak average is around 160 MB/s, ie. a factor of three improvement. In fact we now generally reach throughput rates we were never able to achieve with the old setup.

3 Current Investigations

In early 2018, ATLAS performed various recall tests on tape systems at all T1 centres, the main goal being to check the overall tape performance at various T1 centres. These stress tests help T1 centres to optimize existing setups, identify bottlenecks, and improve their systems in accordance with their needs. At GridKa, ATLAS performed the tape stress test in July

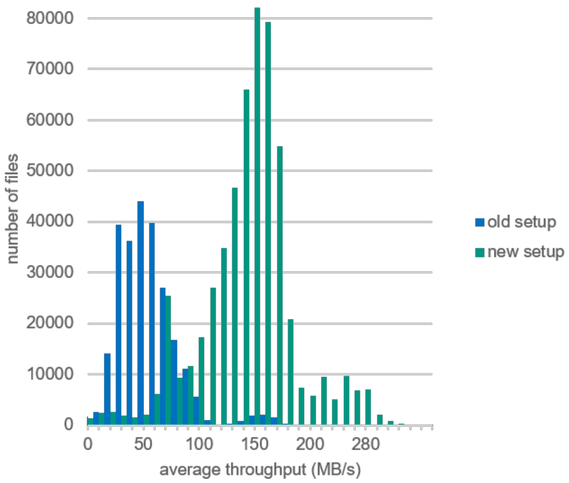

Figure 2: Histogram of Recall Rates per File

2018, recalling 25 datasets with aproximately 170TB of data and 92000 files. For this test 8 T10K-C drives were used. The campain took 8 days with combined recall rate of 400 MB/s.

Figure 3 summarises the total data transferred for each required cartridge. The colour bars indicate the number of required mounts per cartridge. For some cartridges nearly all data on the tape were part of a specific dataset and in an optimal solution we would have accounted for this and simply restored the whole tape, which would have been possible within 6.5 to 7 hours. Instead our dCache system keeps an active queue with a constant 2000 requests. The reason for this behaviour is because of our current infrastructure: we use a legacy tape connection script (tss) [2] between dCache [3] and IBM's Spectrum Protect [4]. The functionality of our dCache to tape connection is illustrated in figure 4. Unfortunately our current setup does not allow a queue length of more than 2000 files, because of a poor memory management.

However our tape system gets a request of 2000 files, which will be sorted by their location on tape. Lets assume these 2000 files are located on 10 different cartridges, and tss is allowed to use only 8 tape drives. Tss builds up 10 tape family queues and 8 of these will be activated to start recalling the files from tape directly into the dCache pool. A queue is sorted by the location of the files on tape and as such prevents other files to be inserted in this queue, once its status is active. Only after one of these active queues is empty will the cartridge be released from the tape drive and a new queue become active. While dCache continuously refills the Recall Queue, tss has no tape drive available and will build up new queues in the background. By the time one queue has been fully recalled there will be new queues build up and the next will be activated irrespective of whether the data in the queue is located on a cartridge which may just have been released from a tape drive. Thus it is necessary to mount the same cartridge over and over again, causing delay since the newly mounted tape has to be rewound to the first file in the new queue. In reality to restore a tape with the maximum amount of data took 34 hours.

After we realised this behaviour we adapted the script and investigated the limits, in order to allow bigger queues and as such an optimised queuing mechanism as can be seen in figure 5. In this newer test we restored the same dataset with an dedicated server and outside of our dCache infrastructure, to keep the environment as simple as possible. The files to be restored have been directly entered in our tss interface [2]. The operating system and the tape connection script on this machine has been specially tuned to support the maximal possible queue length for this host. The length of the new queue was 30.000. As a result, we managed to recall a single tape with one to three mount sessions. In comparison with the 34 hours the restore was reduced to 11 hours, much closer to the minimum time needed to read the whole tape. This is already quite an achievement.

Another aspect we clearly see from this figure, is the amount of tape cartridges used for this dataset. We could have improved the read rate if the data were written to much fewer cartridges in the first place. However this needs more collaboration with the users who wrote the data, since we do not have an idea of the existing dataset boundaries.

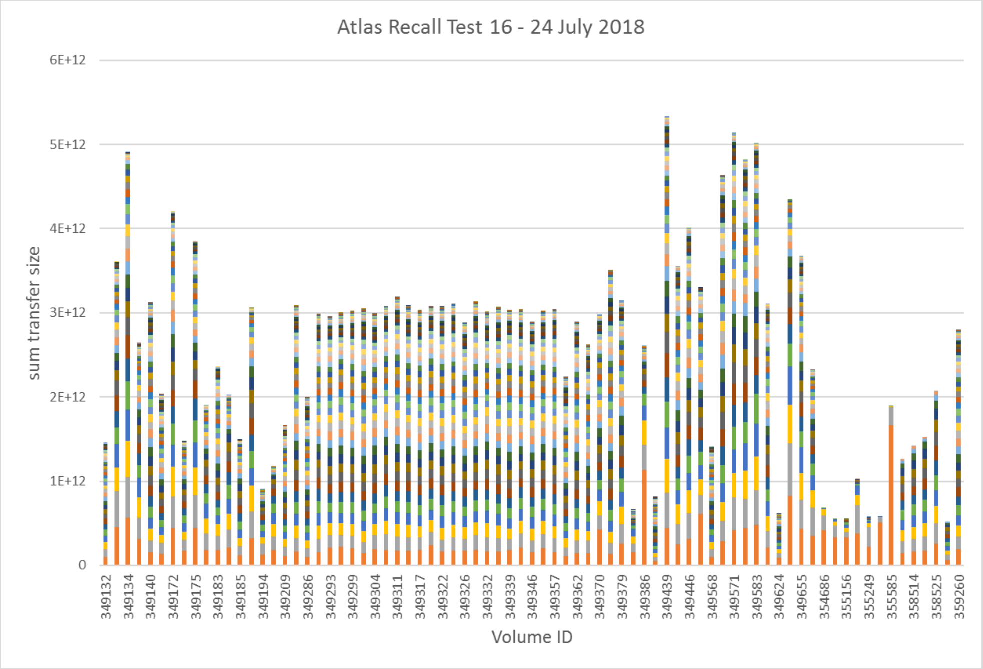

Figure 3: total required data transfer per tape catridge to restore 25 ATLAS datasets

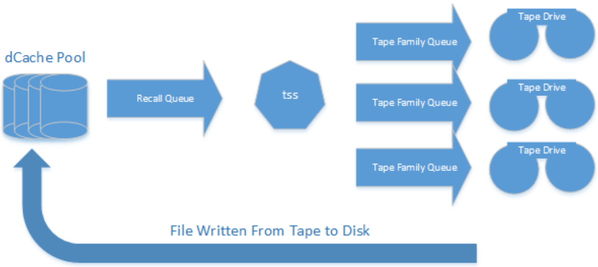

Figure 4: Queuing System from dCache to Tape

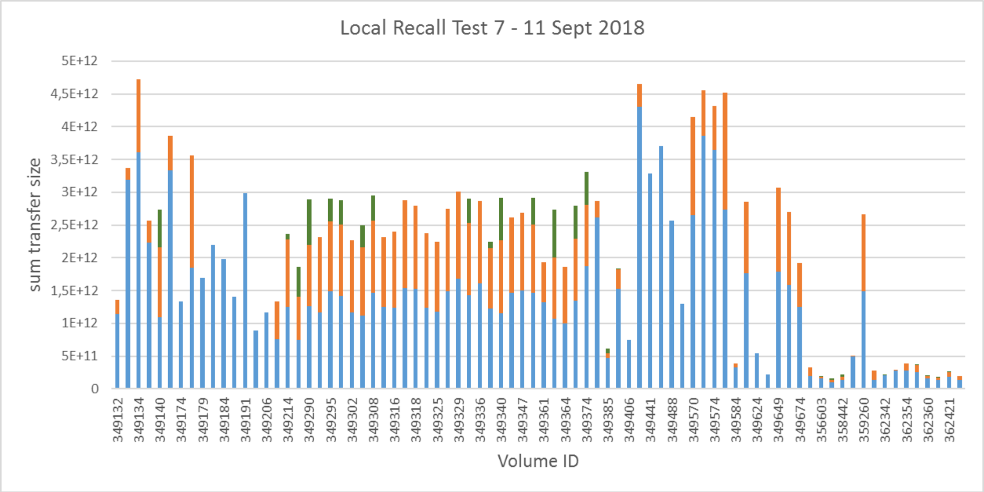

Figure 5: optimized recall test of 25 ATLAS datasets

4 Monitoring

In a complex data-storage system there are many components that need to be continuously checked so that in the case of problems, remedial action can be taken as soon as possible. This presupposes good monitoring tools which include status checking, performance tuning, troubleshooting, and others.

The reason why monitoring is so important will be demonstrated with an example. Typical scenarios which lead to data loss . . . .

In the event that a file cannot be read from tape any longer, the cause of the problem is manifold: the most obvious cause would be that the cartridge itself is defective. Alternatively the cartridge itself may function correctly but the file has not been correctly written to tape. The reason for this could be either, that the drive which wrote the file is broken, or maybe a silent data corruption like a bit flip occurred. If the drive writing the file is broken, it is almost certain that the file will be lost on that tape. To prevent such a scenario, once a file is written to tape it is usual practice that the same tape drive will read it immediately afterwards. If this is successful, it is assumed that other tape drives are able to read the file as well. However we once had a case, that one particular tape drive wrote the data with insufficient magnetization. This drive was able to read the file afterwards, but other drives were unable to read them. We had such a case last year with an Oracle drive. By the time we figured this out, this tape drive had been replaced due to another error. This led to the loss of 4270 datafiles. We are still checking our inventory to discover if there are any more files affected by this cause.

Another scenario would be that the file has been corrupted on an intermediate disk and the checksum of the file has only been built after the file has been written to disk. As such the original file is incorrect. To avoid such a scenario, an end to end data protection mechanism has to be in place. This would need to compare the original file at the remote site on disk with the final version on tape.

However if alarms are generated for every single read-error on a particular cartridge, you might be busy to follow these failures, however if you ignore these failures you will miss a serious cause. On the other hand, establishing a connection between cartridge failures and drive failures might lead quickly to the source of a problem.

In order to avoid these kind of failures the monitoring system needs to be quite complex. GridKa, has several local monitoring tools that contain more specific information about our tape system, such as cartridge names, timestamps for data that were read to and written from tape, the number of retries, mount times, etc. Our local monitoring tools include many components which will be described separately.

Apart from local monitoring tools, we also use established analysis tools, such as Grafana [5] and Elasticsearch/Logstash/Kibana stack [6].

4.1 Grafana

Grafana is a time series analysis software. Through Grafana, it is easy to detect system problems. It allows better understanding of the system from the inside and has a nice feature. such as creating and sharing a dashboard. GridKa Grafana page is accessible from outside and is provided in reference [5].

4.2 ELK-stack

ElasticSearch is a distributed search engine which provides real-time data collection capabilities and integrated features for automatic scaling and replication. For data visualization, we use Kibana, which allows creation of various dashboards very easily. Visualization in Kibana is geared towards the needs of human operators, with various types of bar charts, line charts, timeseries charts, tile maps, etc. So far, our ELK-stack setup is used only internally.

5 Future Developments

In the short term we are currently migrating all our T10KC tapes to T10KD tapes. We have two reasons for this: firstly we gain more capacity and secondly we are able to check the data integrity of all C tapes. In section 4 we described an incident with one C drive which remained undetected for some time. We would now like to know if there are more any sleeping dogs in our system. Once this migration is done, we are ensured that we do not have any further data loss due to this erroneous drive.

In addition we have to re-evaluate the LTO technology. As Oracle has announced that there is no further development in its enterprise technology, it is clear that our current setup with Oracle's enterprise drives cannot be continued for more than five years. Currently we could fill our existing libraries with the existing Oracle enterprise drives, but sooner or later this option will disappear. As a result we have to switch back to LTO technology or abandon our Oracle libraries. If we still believe that LTO will not be an option, we could use IBM's enterprise technology, which is already in place at KIT.

Currently we are using Spectrum Protect for our GridKa data. Our plan is to change this to HPSS (High Performance Storage System [7]), which is a proven technology for scalable environments.

This technology will help us to be prepared for the increasing data volumes in Run 2 and Run 3 of the LHC and the changing access patterns due to evolving computing models. Furthermore we will need to continue our active work together with the experiments on their computing models and pilot setups to prepare for the central European HL-LHC data hub.

5.1 Outreach and Collaboration

We are very active in the community, with regular participation at conferences and workshops. Annually the HPSS users are meeting at the HPSS User Forum (HUF). Twice a year the HEPiX takes place. We also have regular face to face and phone meetings with the WLCG archival storage group. And of course we exchange our experience with other tape storage administrators at Universities within the State of Baden Württemberg.

References

[1] BWDataArchive www.rda.scc.kit.edu
[2] J. van Wezel, D. Ressmann, TSM as tape storage backend for disk pool managers, HEPiX Spring Meeting, Roma, I, April 3-7, 2006
[3] Software dCache https://www.dcache.org/
[4] IBM's Software Spectrum Protect https://www.ibm.com/de-de/marketplace/data-protection-and-recovery
[5] Grafana https://grafana-sdm.scc.kit.edu
[6] ELK-stack https://www.elastic.co/de/elk-stack
[7] IBM's Software HPSS http://hpss-collaboration.org